Remarks/Arguments

Office Action Summary

Status.

1. This *RESPONSE C* is in answer to the Office communication mailed 12/26/2006.
2. a) The Office communication is final.
3. NA

Disposition of Claims.

4. Previous Claims 1 - 6 and newly added Claims 7- 9 are pending in the application.
5. No Claims have been allowed.
6. Claims 1 - 6 are rejected.
7. NA
8. NA

Application Papers.

9. NA
10. NA
11. NA

Priority under 35 U.S.C. § 119.

12. NA

Attachment(s)
NA

DETAILED ACTION

1. Claims 1-6 as amended remain in the application and Claims 7 – 9 are newly added.
2. The Examiner's comment is noted.
3. Claims 1-6 are currently amended.
4. The Examiner's rejection and response is noted.

Response to Applicant's Remarks on Claim Interpretation

5. The Examiner's comments are as follows:

5.1. The Examiner's comment to Prior Remarks 5.1 and 5.2 is as follows:

> *Applicant has repeatedly said that his remarks were wrongly (mis)stated. In response to Remarks 5.1 & 5.2 applicant stated that examiner seems to miss the point that information stored in "PSW and control registers in IBMSl390 architecture" may be inaccurate at one point in time but not accurate in at a later point in time. Examiner has considered the remark however this argued limitation is not presented in the claim language. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., spacial or temporal accuracy of information in PSW & control register) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 11 81, 26 USPQ2d 1057 (Fed. Cir. 1993).*

5.2. Applicant's response to the Examiner's comment to Prior Remarks 5.1 and 5.2 is that Claim 1 and Claim 6 have now been amended to make clear that the legacy state information is compared with the stored state information for each emulation of legacy instructions when legacy instructions have been translated. It is now believed that the Examiner will find the features relied upon by the Applicant are in the claims.

5.3. The Examiner's comment to Prior Remarks 5.3- 5.5 is as follows:

> *In response to Remarks 5.3-5.5, arguments presented are noted by examiner, that the "at a subsequent time for re-execution of translated code, there is no guarantee that the correct "program mode of operation" is stored in the "PSW and Control registers". However, the claim neither states the limitation regarding "at a subsequent time of re-execution" nor it states any limitation regarding guarantee/or lack there for values stored in "PSW and control registers".*

5.4. Applicant's response to the Examiner's comment to Prior Remarks 5.3 - 5.5 is that Claim 1 and Claim 6 have now been amended to make clear that the legacy instructions are emulated multiple times and that *legacy state information* is compared with the *stored state information* for each emulation of legacy instructions when legacy instructions have been translated. It is now believed that the Examiner will find the features relied upon by the Applicant in the claims.

5.5. The Examiner's comment to Prior Remarks 5.6 is noted.

5.6. The Examiner's comment to Prior Remarks 5.7 is noted

5.7. Applicant's response to the Examiner's comment to Prior Remarks 5.7 is that Applicant argued in Prior Remarks 5.7 that in the cited reference (U.S. Patent No. 5,577,231 to Scalzi) "*at the time of execution of the translated instruction, the required information is nowhere reliably stored*". Further, in Scalzi, the "*required information is not guaranteed to be present when it is needed*" With the present amendments to Claim 1 and Claim 6, the explicit storing of the required *stored state information* is recited so that it is explicitly known at the time of execution of the translated instructions.

Response to Applicant's Remarks for 35 U. S. C. 5 102

6. The Examiner's comment is that "*Claim 1-3 and 5 were rejected under 35 U.S.C. 102(b) as being anticipated by Scalzi.*"

6.1. Regarding Claim 1 and Prior Remarks 7.1, the Examiner further noted Applicant's argument that "*Scalzi does not store state information with translated instructions, but rather creates an entirely new VEA address location for every different change in state information.*"

6.1.1. In response to Applicant's Prior Remarks 7.1 the Examiner argues:

In response to Remarks 7.1, examiner respectfully disagrees that claimed limitation precludes teaching of Scalzi. Scalzi teaches legacy information translation is known in the art, however lack of state information (Co1.4 Line 46-Co1.5 Line 51) is augmented with storing the state information with the translation on per instance basis. State information is taught as "source access authorization" in this section. Examiner would further like to point out to Scalzi Co1.6 Lines 22-30.

> *This invention enables an executing processor's DAT mechanism to translate a source DAT ON or OFF page into a plurality of target virtual pages which may be translated to the same page frame in real storage. Each target virtual page is used to represent one combinatorial set of access authorization states existing during one access of a translated page frame in real storage. All target virtual pages translating to the same target page frame (in real storage) may herein be called "synonym" source virtual pages.*

Hence Scalzi teaches the claimed limitation.

6.2. Applicant's response to the Examiner's comment to Prior Remarks 7.1 is that the Examiner apparently does not disagree with Applicant's argument (summarized in Section 6.1 of Current Remarks) that "*Scalzi does not store state information with translated instructions, but rather creates an entirely new VEA address location for every different change in state information.*" The Examiner, however, argues (as quoted in Section 6.1.1 above of Current Remarks) that "*lack of state information (Col.4 Line 46-Col.5 Line 51) is augmented with storing the state information with the translation on per instance basis.*"

6.3. Applicant's further response to the Examiner's comment to Prior Remarks 7.1 is as follows:

6.3.1. The creation of a new VEA address for every change in state information as done in *Scalzi* is not the same as storing *stored state information* along with storing corresponding translated instructions as done in the present invention.

6.3.2. The Examiner concludes that *Scalzi's* operation *is augmented with storing the state information with the translation on per instance basis.* This conclusion by the Examiner does not accurately describe *Scalzi's* operation. More accurately, *Scalzi's* operation is the creation of a new VEA address for every change in state information and storing the translation at the new VEA. This new VEA address creation leads to the need for a very large memory size in the target system as admitted by *Scalzi.*

6.3.3. By way of distinction, Applicant's translation and storage addressing is done independently of the state information and is coupled with explicit storage of the *stored state information* along with the translated instructions in an explicit state word location. Applicant avoids a new address (new VEA address in *Scalzi*). For Applicant, when a new emulation of previously

translated instructions occurs, whether the new *legacy state information* for the new emulation is the same as or different from the previously existing *stored state information*, the new *legacy state information* in a query is compared with the *stored state information* (stored with the previously translated instructions) to see if a match occurs. As is apparent, Applicant's address formation for storing translated instructions does not use state information. The translated instructions are stored in the present invention at a stored location irrespective of state information. However, use of the translated instructions at the stored location is determined by the *match* or *no-match* of the *legacy state information* and the *stored state information.*

6.3.4. Applicant's Claim 1 and Claim 6 have been amended to recite the storage of the *legacy state information* as *stored state information* and the querying of the *stored state information* to determine if a match exists between the *stored state information* and the *legacy state information.* Such storage and query is not done in *Scalzi* and would serve no purpose in *Scalzi* because *Scalzi* uses state information to form the VEA address and hence the correlation must exist.

6.4.Regarding Claim 1 and the Examiner's further comments regarding Prior Remarks 7.2 through 7.7, the Examiner sates:

> *In Remarks 7.2, applicant has argued that amount of additional memory used is small compared to the operation of Scalzi. Further the present invention operates independent of whether a DAT facility is present or not whereas Scalzi is only for a DAT mechanism.*
>
> *In response to Remarks 7.2, examiner asserts that it is unclaimed subject matter as to how much memory Scalzi uses or if the DAT is used or not, as these limitation are not precluded from the claimed limitations. See In re Van Geuns, 988 F.2d 11 81, 26 USPQ2d 1057 (Fed. Cir. 1993).*

Further, in reference to translating source instructions and storing them, Scalzi does not limit to mapping the DAT translated pages only, as can be seen from the Scalzi (Col.6 Lines 22-25 - see above), where Scalzi handles both DAT ON and DAT OFF pages.

In Remarks 7.3-7.5, applicant has argued that Scalzi Col.6 Lines 6-59 has no discussion whatsoever of the translated instructions but only discusses addresses and DAT.

In response to Remarks 7.3-7.5, examiner agrees, however the translation is disclosed in the background of Scalzi and Scalzi (Background, Col.4 Line 46-Col.5 Line 51) whereas the cited section builds on the translated code to associate it to the state information to complete the translation, thereby curing the deficiency of the prior art.

In Remarks 7.3 & 7.6-7.7, applicant argued that there is no discussion of storing the translated instructions or storing any state information with such instructions.

In response to Remarks 7.3 & 7.6-7.7, examiner respectfully disagrees as points to Scalzi's disclosure in Background section (Scalzi Col.1-3), where various translation mechanisms like emulation, binary translation etc are disclosed to translate from one architecture to another (RISC to ClSC etc). The cited section builds on the translated code to associate it to the state information to complete the translation, thereby curing the deficiency of the prior art. Scalzi Col.7 Lines 26-31:

> *It is another object of this invention to use extended (exploded) virtual effective addresses in a target machine to represent, in composites, source program access states and source program logical storage addresses for source programs designed to execute under a different computer architecture.*

Scalzi Col.8 Line 58 -Col.9 Line 31(Subsection cited below):

> *The extended target virtual addresses, in which different target addresses are used to address a single source location, one target address for each source storage access state of an access are called "exploded" target virtual addresses. This is because each source address is transformed to one of a very large number of possible target virtual addresses depending on the state*

of the access control variables at the time of execution, for accesses to source machine storage as represented in the target machine storage. All accesses to source real storage made by processes of the target machine in providing source program execution are made using a target exploded effective virtual address.

This invention generates a "target exploded VEA by combining the "source EA and the "source state indicators". The preferred combining technique is to concatenate the "source EA and the "source state indicators" as respective fields in a target register or in real storage. Subfields in "source state indicator field" receive re-defined source state indicator codes representing the current operating state of the access authorization mechanisms for the source program existing when the target exploded address is being generated.

The target architecture may for example be the IBM PowerPC RISC architecture operating in a target microprocessor, which has a 64 bit virtual address. If the source program is using the S/390 architecture's access authorization mechanisms, the S/390 effective address requires 31 bits of the target 64 bit virtual effective address, leaving 33 bits (64-31) of the target virtual effective address for use as an appended part. Then the target virtual effective address is interpreted by the target DAT as the "exploded virtual effective address" of this invention.

A major advantage of this invention using a PowerPC microprocessor (or any processor having a 64 bit virtual address) is that no modification is needed to the hardware to handle this invention, due to the availability of the 33 bits in the target virtual address (beyond the 31 bits in the source effective address) for indicating the source storage access authorization states in the exploded virtual effective address, which is then applied to the DAT feature of the microprocessor.

Although, applicant's reading of the Scalzi is correct about synonym virtual pages, the current claim does not exclude such an association/mapping. Therefore examiner respectfully disagrees

that a wrongful attempt is being made to create. correspondence (In response to Remarks 7.8).

Applicant's argument regarding inherency are considered and are found to be unpersuasive. Rejection is maintained.

6.5. The Examiner's further comments regarding Prior Remarks 7.2 through 7.7 as quoted in Section 6.4 above are directed, among other things, to arguing that the dynamic address translation (DAT) facility of Scalzi is the same as Applicant's claimed invention.

6.5.1. As is clear from Scalzi and the Examiner's quotes in Section 6.4 above, Scalzi uses a DAT facility to create a new VEA for every change in sate information. The Examiner appears to argue that storage at the VEA is storage of the state information used to generate the VEA.

6.5.2. Assuming, for purposes of argument only, that such storage in Scalzi is storage of information that has some correlation to state information, nonetheless Scalzi does not store *stored state information* as a result of an initial emulation which can be used in a query in a subsequent emulation to determine a match with the *legacy state information* from the subsequent emulation when a new emulation as claimed in Applicant's Claim 1 and Claim 6.

6.5.3. The difference between Scalzi and the present invention can be understood with an example. In the example, assume that Scalzi has a first emulation of legacy code having first state information. The Scalzi DAT will generate a first VEA (based on the first state information) and will store translated instructions at the first VEA adddress. Assume then that Scalzi has a second emulation of the same legacy code but having second state information. The Scalzi DAT will generate a second VEA address (based on the second state information) and will store translated instructions at the second VEA

address. In the first emulation, there is no *stored state information* at the first VEA address. In the second emulation, there is no *stored state information* at the second VEA address. In the second emulation, there is no *stored state information* at the second VEA address which can be used in a query to determine a "match" or "no-match" with *legacy state information* that existed for the first emulation. These examples of the operation of Scalzi demonstrate why Applicant's Claim 1 and Claim 6 distinguish over Scalzi, that is because those claims require a query to determine if *stored state information* matches new *legacy state information.*

6.6. Regarding Claim 2 and the Examiner's comments regarding Prior Remarks 7.10.1, the Examiner sates:

> *6.6.1. Regarding Claim 2 Applicant has argued that (Remarks 7.10.1), no translated blocks are stored or state information associated with them. Further questions regarding block boundaries are raised. As shown before the translation whether done at execution time or before hand is associated with the state information (also see Col.7 Lines 26-31, Col.8 Line 58 -Col.10 Line 8) is stored in the virtual pages (Col.6 Lines 6-30). Applicant's arguments that no state information is stored in the blocks, is misleading in view of above teaching. Examiner respectfully maintains the rejection.*

6.6.2. As discussed in Section 6.5.1 above, the Examiner appears to rely upon the argument that storage at the new VEA is inherently storage of the state information used to generate the VEA. Claim 2 as currently amended recites "*said stored state information is stored as a state word in each of said particular translated blocks*". Such storage as a state word is not done in *Scalzi* and would have no purpose to be done in *Scalzi.*

Response to Applicant's Remarks for 35 U.S.C. § 103

7. Regarding Claims 6 and 8 and the Examiner's comments regarding Prior Remarks 8.1, the Examiner sates:

> *Claim(s) 6 & 8 were rejected under 35 U.S.C. 103(a) as being unpatentable over Scalzi, in view of Mann. Regarding Claim 8 (Remarks 8.1) Motivation to combine Mann with Scalzi is clearly defined in the rejection. The issue raised by applicant is how the DAT (disclosed by Scalzi) used with teachings of Mann. The dynamic address translation (DAT) addresses the associating the state information with the translated code, whereas dynamic object code translation (DOCT) taught by Mann is directed to instruction translation from one platform to another. Scalzi does not limit how & when the actual code is translated (one way taught by Mann) as it presents commonly known method in art in background section. Scalzi is concerned with loss of state information, which is pertinent to translated code at the time of translation. Hence, Mann performs translation of the actual code, Scalzi associates the translated code with the state information and stores it making execution of source code on the target machine possible with "source access authorization". Therefore Mann complements Scalzi in performing successful translation.*

7.1. Assuming for purposes of argument that the Examiner's characterization of Mann and Scalzi is correct, the Examiner still has not described how, in fact, Mann and Scalzi could function together. Scalzi requires that different **new addresses** (VEA) be generated as a function of different state information. Mann requires that the **same addresses** be used for emulation (except when changes have occurred and execution is returned to interpretation and hence no addresses are used). Accordingly, the operations of Mann and Scalzi are mutually inconsistent. The Examiner has not explained how the combination of Mann and Scalzi would form an operable system. No one skilled in the art would find it obvious to combine references that have mutually inconsistent requirements (such as **new addresses** versus **same addresses**). The Examiner's proposed combination of mutually inoperable references does not even establish a prima facie case of obviousness.

Claim Rejections - 35 USC § 102

The quotation of paragraphs of 35 U.S.C. 102 (b) are noted.

8. Claim 1-3 and 5 are rejected under 35 U.S.C. 102(b) as being anticipated by U.S. Patent No. 5,577,231 issued to Scalzi et al (Scalzi hereafter).

8.1. Regarding the rejection of Claim 1-3 and 5, the Examiner argues as follows (the underlined portion shows changes in the Examiner's argument compared with the previous Office action):

> *Scalzi teaches (Original) method for dynamic emulation of legacy instructions of a legacy program (Scalzi: Co1.13 Lines 26-41) by*
>
> *(a) providing state information for determining a program execution mode for emulating said legacy instructions (Scalzi: Co1.5 Lines 18-20, Co1.8 Lines 41-45) as source program containing instructions and operating states during emulation of source program;*
>
> *(b) accessing said legacy instructions and said state information (Scalzi: Co1.8 Lines 21-40, 57-61) as current state of the executing source program representing access authorization states defined by the source architecture and for each particular legacy instruction,*
>
> *(c) querying to determine if one or more particular translated instructions for said execution mode are stored as a result of translating said legacy instruction for said execution mode (Scalzi: Fig. 13 also see Co1.9 Lines 32-*

62; Col.4 Line 46-Col.5 Line 51) as one or many effective translated instructions representing source instructions are translated into target instructions represented by their effective addresses, and

(d) if not translated for said execution mode, translating the particular legacy instruction into one or more particular translated instructions for emulating the particular legacy instruction for said execution mode (Scalzi: Col.4 Line 46-Col. 5 Line 51 ; Col. 6 Lines 6-59 - dynamic address translation (DAT) process checking the page table) as known process of emulation being supplemented by state information storage using the dynamic address translation,

(e) storing said one or more particular translated instructions with said state information as storing the target address and state information associated to source code in target virtual address which is associated to translated code (Scalzi: Col.7 Lines 26-31; Col.8 Lines 58 Col.10 Lines 8), and if translated for said execution mode, continuing without additional translating (Scalzi: Col. 11 Lines 12-15) as presence of valid page table entry (PTE) indicates already translated source code with state information and execution continues if valid PTE is found in the PT as detailed in Scalzi (Scalzi: Col.9 Lines 32-62),

(f) accessing said one or more particular translated instructions for emulating said legacy instructions for said execution mode (Scalzi: Col. 11 Lines 44-64) as source

instructions along with storage authority architecture being executed on the target processor architecture.

8.2. Regarding the Examiner's argument relating to the rejection of Claim 1-3 and 5 as quoted in Section 8.1 above, under the "if not translated condition" of paragraphs (d) and (e), the Examiner's argument that "storing said one or more particular translated instructions with said state information as storing the target address and state information associated to source code in target virtual address which is associated to translated code" is not accurate and misleading. No state information is stored at the VEA address, but rather Scalzi generates the address (VEA) based upon the state information. Additionally, under the "if translated condition" of paragraph (e), the present invention requires "querying said *stored state information* to determine "if a match occurs between the *legacy state information* and the *stored state information*". Since Scalzi does not store state information, no query or match determination can occur in Scalzi as required by Applicant's claims. Further in the absence of a match in the present invention, the "translating and storing steps" are performed and such operation would be meaningless in Scalzi. Scalzi cannot have a condition that state information used to form the VEA address does not correlate with the VEA address. The information correlated to the Scalzi VEA address must correlate (match) since Scalzi uses the state information to form the VEA address. The penalty that Scalzi suffers in having every VEA address correlate to satate information (match) is Scalzi's requirement of a very large memory capacity. Scalzi's requires of a very large memory capacity because Scalzi requires a different VEA address for every different instance of state information.

8.3. The present invention by way of distinction does not require the large memory capacity required by Scalzi. Rather than mapping every difference in state information to a different VEA address as done in Scalzi, the present invention permits mapping to the same address. The present invention, however, requires a "match" and "no match" determination to ascertain whether the information stored

at the address can be used. The present invention by querying each emulation for the "match" or "no match" condition is an operation that requires far less memory than Scalzi. The query to detect the "match" or "no-match" condition for each emulation appears in independent Claim 1 and Claim 6 and hence these claims and claims dependent there from are believed allowable.

8.4. Regarding the rejection of Claim 2, the Examiner argues as follows:

> *Scalzi teaches storing of the one or more particular translated instructions is in one or more particular translated blocks and said state information is stored in each of said particular translated blocks as page frames (Scalzi: Col.6 Lines 6-30) as DAT based translation and exploded virtual target effective address (VEA) (Scalzi: Col.9 Lines 32-62; Col.24 Lines 55-65).*

8.4.1. The Examiner's argument as quoted in Section 8.4 above that "*state information is stored in each of said particular translated blocks as page frames*" is not accurate. Scalzi uses *state information* to form addresses, but Scalzi does not store state information at the addresses formed. The Examiner has argued that state information is correlated with VEA addresses since state information is used to generate each VEA address. Even if the Examiner's argument is assumed to be true for purposes of analysis, Scalzi does not operate as required by Applicant's Claim 2 such that "*said stored state information is stored as a state word in each of said particular translated blocks*". Accordingly, Claim 2 is believed allowable.

8.5. Regarding Claim 3, Claim 3 depends from Claim 1 and is believed distinguished for the same reasons.

8.6. Regarding Claim 5, Claim 5 depends from Claim 1 and is believed distinguished for the same reasons.

Claim Rejections - 35 USC § 103

The quotations of paragraphs of 35 U.S.C. § 103 and related information are noted.

9. Claims 4 & 6 stand rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,577,231 issued to Sscalzi et al (Scalzi hereafter), further in view of U.S. Patent No. 6516295 issued to George A. Mann et al (Mann hereafter).

9.1. Regarding the rejection of Claim 4, the Examiner argues as follows:

> *Teachings of Scalzi are shown in claim 1 rejection above. Scalzi is indifferent in teaching that legacy instructions are object code instructions compiled/assembled for a legacy architecture (Scalzi: Col.5 Lines 14-28 & Background - high level translation trivial, binary translation of object code has limited applicability). Although, his indication is clear that high-level source code translation is trivial (Col.2 Lines 1-8) and object code translation though binary translation does not reproduce access-storage-authorization mechanism - state based (Scalzi: Col.2 Lines 27-33). Hence although not explicitly, Scalzi indirectly teaches object code based translation with accompanying state information.*
>
> *Mann explicitly teaches legacy instructions are object code instructions compiled/assembled for a legacy architecture (Mann: Col.2 Lines 44-51).*
>
> *It would have been obvious to one (e.g. a designer) of ordinary skill in the art at the time the invention was made to apply the teachings of Mann to Scalzi. Scalzi is concerned with correct state information porting while performing translation of the legacy information is major concern, source instruction code to target instruction code translation is not a specific concern which is understood as known in art (Scalzi: Col. 5 Lines 26-28). The source instruction code to target instruction code translation teaching is supplemented by teachings of Mann. The motivation to combine would have been that both correct state information with address translation information (taught by Scalzi) and instruction code translation (taught as DOCT by Mann Col.2 Lines 44-51) are absolutely essential for correct legacy S/390 object code to target RISC based code translation.*

9.2. Regarding the Examiner's arguments supporting the rejection of Claim 4, the Examiner's argument for "motivation to combine" is that "*both correct state information with address translation information (taught by Scalzi) and instruction code translation (taught as DOCT by Mann Co1.2 Lines 44-51) are absolutely essential for correct legacy S/390 object code to target RlSC based code translation*". However, neither Mann nor Scalzi recognized such necessity. The Examiner appears to be relying only on Applicant's disclosure to find such a teaching and hence no proper motivation to combine has been shown by the Examiner.

9.3. Furthermore, the Examiner still has not, as requested by Applicant in the prior *RESPONSE B*, described how Mann and Scalzi can function together. Scalzi requires that different **new addresses** (VEA) be generated as a function of different state information. Mann requires that the **same addresses** be used for emulation. If no emulation is possible in Mann, execution is returned to interpretation. Accordingly, the operations of Mann and Scalzi are mutually inconsistent, one requiring **new addresses** and the other requiring the **same addresses**. The Examiner has not explained how the combination of Mann and Scalzi would form an operable system. No one skilled in the art would find motivation to combine references that have mutually inconsistent operation.

9.4. Regarding the rejection of Claim 6, the Examiner argues as follows:

> *Claim 6 discloses similar limitations as claim 1 and is rejected for the same reasons as claim 1. Preamble of claim 6 presents added limitation related to legacy code to be in object code format of the source. Mann shows this teaching in claim 4 rejection above.*

9.5. Regarding the Examiner's arguments supporting the rejection of Claim 6, the Applicant's arguments with respect to Claim 4 in Sections 9.2 and 9.3 above are believed to equally apply to Claim 6. Accordingly Claim 6 is believed allowable for the same reasons as discussed in connection with Claim 4.

10. The newly added Claim 7 through Claim 9 are dependent from Claim 6 and are therefore believed allowable at least for the same reasons as Claim 6.

Respectfully submitted,

	SIGNATURE	
David E. Lovejoy (US Reg. No.: 22,748)	David E. Lovejoy **/david lovejoy/**	**Signature Date** May 14, 2008
Mail Address	Customer No.	Communication
102 Reed Ranch Rd. Tiburon, CA 94920-2025 USA	21603	Tel: (415) 435-8203 Fax: (415) 435-8857 e-mail:david.lovejoy@sbcglobal.net